Intel Developer Update is Intel's monthly online news magazine for developers. As the official publication of developer.intel.com, it brings hardware, software, and Web developers the latest information on Intel initiatives, technologies, platforms, and products.

Cover Story

Each month, we run a cover story on the most significant industry announcement, trend, or development for the month.

Featured Articles

Delivering in-depth reports on key platforms, products and technologies, our featured articles provide a monthly source of information on issues affecting developers. Be sure to check in every month for the latest developments driving the evolution of the industry.

Contact the Editor

To make *Intel Developer Update* a better information resource, we invite you to share your thoughts on what we've published or what you'd like to see covered. Comments are always welcome.

Archives

Our archives contain two groups of previously published articles. One group contains all the articles that appeared in *Platform Solutions News*, the earlier version of *Intel Developer Update*. The articles date from September 1997 through August 1999. The other group is set up to contain *Intel Developer Update* articles dating from the inaugural September/October 1999 issue.

Bookmarking

We advise against bookmarking article pages. They're accessible online only during the month the issue is live. Thereafter, they're removed to our archives. Instead, we suggest that you bookmark the PDF (Adobe* Portable Document Format) file versions of the articles. You'll find buttons for the PDF files labeled "print article" in the right navigation section of each article. A PDF for the entire issue is labeled "print magazine" and is located near top right side of the IDU home page.

Table of Contents

(Click on page number to jump to articles)

Cover Story

Micro Signal Architecture: More than a DSP

David Borland
Manager
DSP Design Center
Intel Corporation

Overview

Intel Corporation and Analog Devices, Inc. (ADI) have introduced the new "Micro Signal Architecture," a new and highly integrated digital signal processor (DSP) architecture with integrated microcontroller functionality that enables significant improvements in performance, power consumption, and ease of programmability in battery-powered communications applications.

Micro Signal Architecture was developed jointly by Intel and ADI, which began working on this brand-new architecture in February 1999. A development team of engineers from both companies was headquartered in Austin, Texas, and development work was also done in Chandler, Arizona, Norwood, Massachusetts, Bangalore, India, and Edinburgh, Scotland.

The new highly integrated Micro Signal Architecture they created integrates DSP and microcontroller functionality. Instead of relying on dedicated DSPs for audio and video tasks and a separate microprocontroller for control, a single core architecture can now handle both DSP tasks and simple control functions, eliminating the need for additional system processors.

Intel and ADI will develop separate products based on this architecture core. The Micro Signal Architecture core will initially operate at up to 300 MHz. A roadmap is in place leading to 1-GHz performance in the future. The core features a 64-bit core memory bus that supports up to 3.2 Gbytes/second at 500-MHz bandwidth for internal data accesses.

The core is designed to operate at less than 1 volt, enabling AA battery-powered products to support rich multimedia functions such as streaming video, voice, and handwriting recognition. The core integrates a sophisticated Dynamic Power Management system that can scale power dissipation according to the signal-processing load, delivering up to 10 times the battery life of present-day designs.

Micro Signal Architecture can help trim weeks from software development time. It incorporates a simplified programming model based on an advanced compiler that enables programmers to build signal processing and control programs using large blocks pre-written in C/C++ code.

Micro Signal Architecture is designed to work in combination with memory, serial ports, mixed signal circuits, external memory interfaces, and peripherals to support a variety of general-purpose DSP products and application-specific chipsets. It will play a central role in the Intel® Personal Internet Client Architecture, designed to accelerate the development of next-generation wireless Internet access devices.

Next-Generation Wireless Requirements

Computers and communications devices are experiencing a functional convergence as consumers demand more sophisticated technology to deliver richer content and more robust applications on wireless client devices.

At the same time, the industry is increasingly focused on Internet enabled, wireless mobile devices, including 3G cellular phones, two-way pagers, Telematics (automotive), and sophisticated PDAs. These devices can include a variety of processor-intensive signal processing tasks including text-to-speech, speech-to-text, handwriting recognition, and rich full-motion video.

In addition to DSL and cable modems, emerging technologies including Bluetooth* wireless connectivity, GPRS (General Packet Radio Service), and 3G cellular are increasing the quantity of data that can be delivered to mobile client devices. This increased bandwidth will support the transmission of much richer multimedia content, including high-quality audio, video, and images.

The Micro Signal Architecture is designed to meet these requirements by integrating DSP capability and microcontroller functionality in a single low-power core architecture supported by simplified programming model.

Computational Units

DSP performance for video, image, voice, and data in communications tasks depends on robust and flexible computational units. Every core in the Micro Signal Architecture family contains MAC (multiply accumulate), ALU (arithmetic logic unit), and shifter functions, with the number and mixture of these resources variable from one processor core to another. The first Micro Signal Architecture implementation features dual MACs, dual ALUs, and a single-barrel shifter.

- *The MAC operation* is one of the most frequent operations in multimedia and communications applications. In the initial Micro Signal Architecture implementation, each MAC is capable of performing one 16-bit by 16-bit multiply per cycle, with an accumulation to a 40-bit result.
- *The 40-bit ALUs* operate on 8-bit, 16-bit, 32-bit, or 40-bit data. In a single cycle the ALUs can execute two 32-bit operations, or four 16-bit operations, or four 8-bit operations.
- *The 40-bit shifter* with extensive capabilities for performing shifting, rotating, normalization, extraction, and depositing data.

MACs or ALUs can be used in the same instruction. The data for the computational units is found in a multi-ported 16-entry, 32-bit register file.

Easier Programming

The Micro Signal Architecture has been optimized for programming both the DSP and the application control code in C/C++. After compilation, the code can be reviewed with advanced profiling tools to define areas that have the most intense signal processing requirements.

These areas can then be replaced with assembly code through the use of library functions, DSP benchmarks, or handwritten code developed from the processor's assembly-level instruction set. This instruction set follows an algebraic syntax that presents the programmer with an extremely readable code.

As a result, in most applications approximately 80 percent of the code can remain in C/C++, which substantially reduces the effort required to optimize the key elements of the algorithms.

An integrated performance monitoring unit supports event counters for analysis of cache hit rates and other events. A debug unit provides six address and two data breakpoints plus a 16-entry trace buffer to assist in program debug.

Multifunction Instructions

To minimize memory usage, the instruction set supports 16-bit op-codes to represent the most frequently used instructions in addition to 32-bit and 64-bit op-codes for multifunction instructions. These combinations result in excellent code density, while allowing the programmer to combine multiple resources in a single instruction. The architecture features an enhanced media instruction set. SIMD (single instruction multiple data) instructions operate on up to four bytes of data in one cycle.

The dual ALUs capable of operating on 8-, 16-, 32- or 40-bit data provide the flexibility to cover the signal processing needs of a wide spectrum of application requirements. By viewing the registers as pairs of 16-bit operands, dual 16-bit or single 32-bit operations can be accomplished in a single cycle. In addition, the second ALU enables quad 16-bit operations to accelerate per-cycle throughput.

Dynamic Power Management

The new Micro Signal Architecture is the first DSP architecture to feature Dynamic Power Management, which can deliver 10 times the normal battery life at one-third peak performance. Dynamic Power Management uses RTOS and power management logic to continuously optimize performance and power consumption by varying the frequency and the applied core voltage.

The result is power consumption and performance optimized for real-time applications. For example, Dynamic Power Management adjusts the power up for video, balances performance against power consumption for voice applications, and adjusts performance down to maximize battery life in standby mode.

Memory Management

The Micro Signal Architecture includes a 16-Kbyte instruction cache, a 32-Kbyte data cache, and a 4-Kbyte data scratchpad SRAM. A Harvard Architecture works in combination with a Hierarchical Memory Structure:

- *Level 1 (L1) memories* are closest to the core. The L1 instruction memory holds instructions only. The two L1 data memories hold data, and the dedicated scratchpad data memory stores stack and local variable information.
- *Level 2 (L2) memories* include other memories, on-chip or off-chip. L2 features a single unified 4-Gbyte memory space for both instructions and data.

The L1 instruction memory and L1 data memories can be configured as either Static RAMs (SRAMs) or caches.

Instruction and data memory management units (MMUs) provide memory protection for individual tasks that may be operating on the core and can be used to protect system registers from unintended access.

Development Plans

The announcement of Micro Signal Architecture marks the successful completion of the joint development agreement between Intel and ADI announced in February 1999. Core silicon is planned for release to development partners in early 2001. While products based on the core will be developed and marketed separately by ADI and Intel, products from both companies will be compatible at the assembly source and binary code levels.

The new Micro Signal Architecture will play a vital role in the Intel Personal Internet Client Architecture, designed to accelerate the development of next-generation wireless Internet access devices. Personal Internet Client Architecture provides the building blocks for cellular phones, including Intel® XScale™ microarchitecture and flash memory. It will enable Intel to offer solutions to support next-generation wireless devices that integrate memory, computing, and communications subsystems.

Summary

The proliferation of next-generation mobile devices requires a low-cost, low-power architecture optimized for high-bandwidth wireless data transmission. Traditional microprocessors deliver the performance to support data-rich applications, but at the significant disadvantages of higher cost and power consumption, shorter battery life, and greater heat dissipation.

The solution is a DSP architecture optimized to process real-world signals in real time. Such an architecture handles huge streams of images and sounds as well as the text and pure data that Internet access requires, and consumes a fraction of the power.

Designed for the next generation of battery-powered Internet access devices, Micro Signal Architecture supports the low-power consumption and high-performance processing for a wide range of signal processing applications including video, image, voice, and data in communications tasks. This new architecture goes far beyond traditional classical DSPs by combining a highly efficient computational architecture with features such as optimizations for high-level language programming and memory protection, which are usually seen on microcontrollers.

Micro Signal Architecture is ideal for a variety of battery-powered devices that must support high-intensity signal processing on a strict power budget. Its Dynamic Power Management is projected to deliver 10 times the battery life of devices without this capability, and it has a performance roadmap to over 2,000 MIPs.

More Info

Developers can access specifications and development tools for the new core architecture through the ADI-Intel Joint Development Web site.

In the coming weeks and months, developers and additional third parties will be prototyping various applications for the Micro Signal Architecture. To keep current on the latest developments, register for information updates with the ADI-Intel Technology Forum.

Author Bio

David Borland is manager of the Intel DSP Design Center and co-manager of the Joint Development initiative for Analog Devices and Intel Corporation.

David has 16 years of semiconductor industry experience. He joined Intel in 1999 in Austin, Texas, to start the Intel DSP Design Center and joined the leadership team of the Joint Development with Analog Devices. Before working at Intel, David was the product line engineering manager for Cordless, ISDN, and Modem products at Advanced Micro Devices' Communication Products Division. He studied electrical engineering at Louisiana State University and holds a Bachelor of Science degree.

Column

From the Editor
Donna Loveland
Managing Editor
Intel Developer Update Magazine
Intel Corporation

Column

Throughout most of the world, this is the season for holiday celebrations and gift giving. Our gift to you is the new Reader Meter, an online poll for rating IDU articles. You can register your opinion to let us know what interests you, and check live poll results to see what other developers think. If you're in a giving mood, give it a try!

We hope these five articles from IDU will brighten your professional life this season.

Micro Signal Architecture: More than a DSP–*cover story***–**The newly announced Micro Signal Architecture, jointly developed by Intel and Analog Devices, goes far beyond traditional digital signal processing by combining a highly efficient computational architecture with features usually seen in microcontrollers.

Intel® Active Monitor–Thermal Assurance for End Users–This new alerting utility for Intel® Desktop Boards monitors critical parameters and displays important system information. It can provide extra assurance for end users and even lower support costs for integrators.

Intel and HP Debut Deep Forest* Concept PC–Co-developed by Intel and HP, this desktop system features an Intel® Pentium® 4 processor running at 1.5 GHz, seven USB ports, a low-profile AGP4X slot, 10/100 Ethernet, and a low-profile CNR card slot, all packed into a chassis the size of a child's lunchbox.

Intel® ACPI Component Architecture ASL Compiler–The Intel® ASL Compiler provides full support for the new ACPI 2.0 specification and the new ASL elements. It gives BIOS developers a fully functional tool that supports all grammar elements and provides greatly enhanced error checking over existing implementations.

Embedded Intel® Architecture Caching Solutions–Caching is a technique for keeping frequently accessed information closer to the requester. Adding cache appliances based on embedded Intel Architecture to an existing network at critical points can reduce network latency and improve access efficiency while keeping WAN implementation costs under control.

Be sure to use the links in the "More Info" section of each article. They lead to details on these significant topics.

Enjoy.

Author Bio

Donna Loveland is the editor of *Intel Developer Update* magazine. She joined Intel's Platform Marketing group in 1999 as the editor of Platform Solutions News. Donna began her career with Intel in 1982 as a technical editor in an advanced microprocessor development group. Since then, she's held technical and marketing positions in leading-edge technology areas ranging from stereoscopic display to digital broadcast to scalable online content. Donna has a B.A. degree in English from the University of Rochester and an M.A. in Expository Writing from the University of Iowa.

Departments

Desktop

Intel® Active Monitor–Thermal Assurance for End Users

Scott Bair
Senior Marketing Engineer
Desktop Boards and System Marketing
Intel Architecture Group

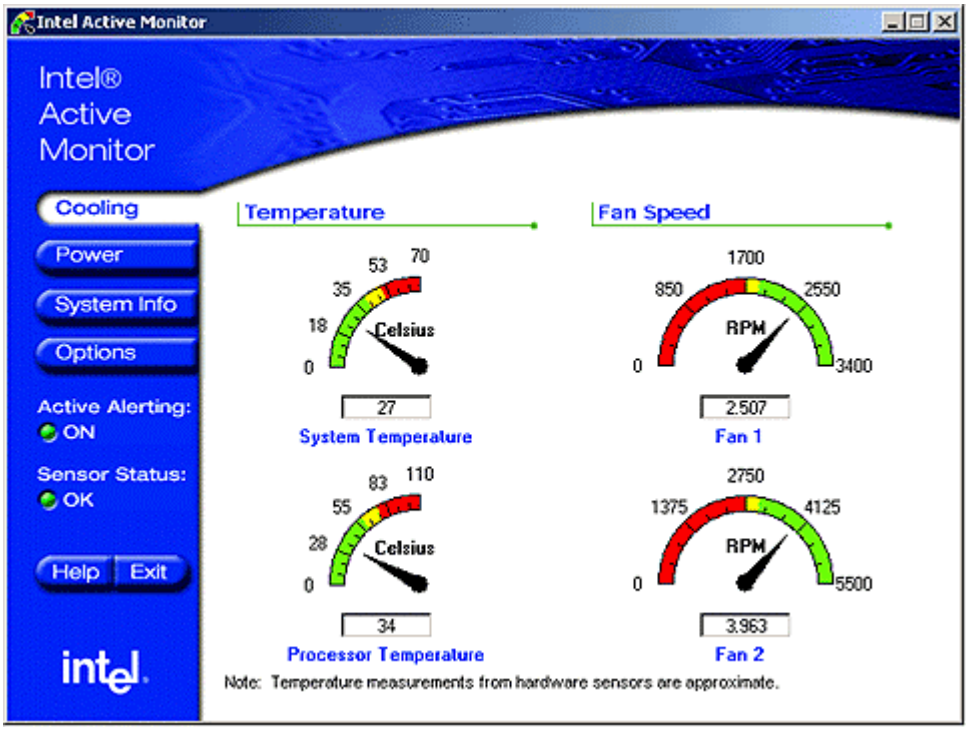
Overview

Tech enthusiasts and power users have long known that faster systems require increasingly better cooling solutions. As processors cross the gigahertz barrier, however, even average PC users are becoming aware of thermal issues. One way that system integrators can reassure their customers is to offer solutions that educate and help protect against failures. The Intel® Active Monitor, a new alerting utility for Intel® Desktop Boards, monitors critical parameters and displays important system information. It can provide extra assurance for end users and even lower support costs for integrators.

Intel Active Monitor

Intel Active Monitor is an Intel developed utility that monitors a system's vital signs. It warns users if their processors or systems begin running too hot or if fans or power supplies are failing. The utility takes advantage of specialized hardware sensors on new Intel Desktop Boards, based on Intel® 810, 815, and 850 chipsets, to provide a virtual "look inside" the system.

The Intel Active Monitor resides neatly on the system tray, using less than 1 percent of system resources. If queried by the user, an attractive GUI pops up. Status "gauges" display processor and system temperatures, power supply voltages, and cooling fan speeds. The side panel shows the sensor and alert status as well as providing access to system information and options. See Figure 1.



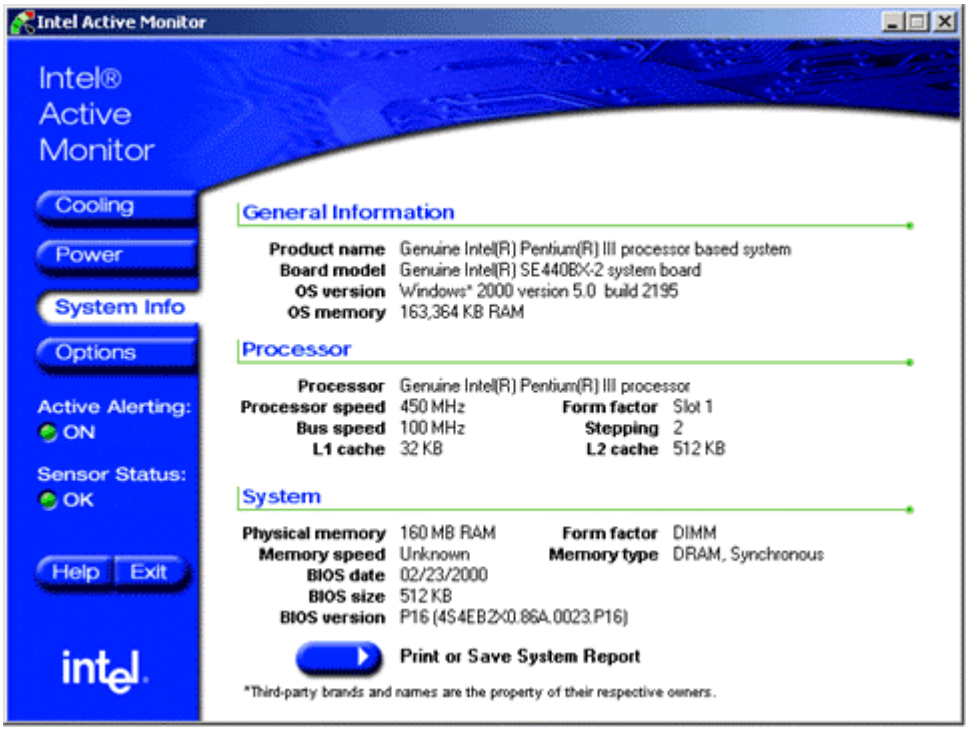
An Ounce of Prevention

Should a failure actually occur, the Intel Active Monitor alerts the user and offers guidance. The user can choose several different styles of alerts: a pop-up window, a blinking “red” tray icon, an audio alert or all three. The GUI takes the user to the gauge that caused the alert. It directs the user to a list of possible causes and recommended solutions through its context-sensitive help. See Figure 2.

The Intel Active Monitor can also serve as a valuable post-sales support tool and can even lower support costs. A support technician can quickly check the current status of a system or view a log file of recent alert activity. By keeping constant watch on the system, the monitor can provide the first warning of a developing problem and help reduce thermal-generated support costs. A system fan, after all, is a lot less expensive to replace than a processor.

Not Just a Pretty Face

The Intel Active Monitor helps educate users about their systems. Its dynamic measurement of temperature and fan speed allows them to virtually look inside their systems. The System Info button lets the user identify processor type and speed, the desktop board model, chipset and memory information, and other details–including System Management BIOS information that cannot be obtained by simply searching through Control Panel. See Figure 3.



The Intel Active Monitor's GUI gives easy access to program options and threshold settings. Anyone can easily customize the alerts and their threshold values. The Intel Active Monitor, however, carefully sets the default values based on Intel-tested hardware specifications. It automatically detects the processor type and sets appropriate thresholds for that specific processor. System Integrators will know that the thresholds are correctly set for any system they produce based on a new Intel Desktop Board.

Where Can I Get It?

The best way to experience the benefits of the Intel Active Monitor is to try it yourself.

First, make sure that you have an Intel Desktop Board equipped with hardware sensors. These sensors are included on most boxed Intel Desktop Boards. Be sure to check your individual board's product spec for availability.

Second, install the software. You can find the utility on the CD that is included with all boxed Intel Desktop Boards. Or you can download the latest version from Intel's Web site. The application has been validated for use with Intel Desktop Boards based on the Intel 810, 815, and 850 chipsets.

Summary

Assure your end users of the effectiveness of your system design by providing the Intel Active Monitor on their desktop. It monitors system temperatures, fan speeds, and power supply voltages on all new Intel Desktop Boards and helps reduce thermal-related support costs. The Intel Active Monitor provides the user with important information at any time and peace of mind at all times.

More Info

The Intel Active Monitor is validated on Intel Desktop Boards based on Intel 810, 815, and 850 chipsets. For more information on features and limitations, please refer to the product documentation.

For more information on the features and benefits of buying Intel Desktop Boards, visit the Desktop Boards area of Intel's Developer Site.

For more information on other software products included on the Intel Desktop Board CD, visit the Intel Boxed Desktop Boards area of Intel's Channel Web Site.

Author Bio

Scott has a bachelor's degree in electrical engineering and an M.B.A. from Brigham Young University. In his three years with Intel, he has contributed to the Intel Desktop Board CD, Intel® Express Installer, Intel Active Monitor, and the Intel® AnswerExpress℠ Support Suite. This is his first appearance in Intel Developer Update Magazine.

Intel and HP Debut Deep Forest* Concept PC

Russ Campbell
Senior Technical Marketing Engineer
Desktop Architecture Lab
Intel Corporation

Overview

What has an Intel® Pentium® 4 processor running at 1.5 GHz, seven USB ports, a low-profile AGP4X slot, 10/100 Ethernet, and a low-profile Communications and Networking Riser (CNR) card slot, all packed into a chassis about the size of a child's lunchbox?

The answer is the Deep Forest* Concept PC, a prototype platform co-developed by Intel and Hewlett-Packard Co.

Intel welcomes opportunities to collaborate with PC OEMs to explore the future possibilities of PC computing. In the case of Deep Forest, these possibilities include a Concept PC with the processing power to handle the graphics demands of the Visual Internet, multimedia, and e-Business applications.

While the Pentium 4 processor will initially be used in high-end systems, the Deep Forest Concept PC demonstrates that Intel's highest level of computing power can be delivered in a small-footprint, legacy-reduced, expandable PC platform.

"Deep Forest is a collaborative effort to address a new category of desktop computing that is powerful, easy, small, legacy-reduced, and network-optimized," says Pat Gelsinger, vice president and chief technology officer, Intel Architecture Group.

10" x 13" x 4"

The Deep Forest Concept PC features a petite footprint that measures approximately 10" x 13" x 4," about the size of a lunchbox.

Legacy hardware features, including internal PCI expansion slots, as well as serial and parallel ports, have been removed in order to achieve a small system footprint and to support "Ease of Use" expansion opportunities. The Intel® Deep Forest Concept motherboard supports the Intel® 850 chipset and an additional USB controller to support the platform's seven USB ports. This high level of connectivity can accommodate flexible external expansion options including a keyboard, mouse, DVD, external storage devices, graphics tools, and other peripherals.

The Deep Forest platform supports interface specifications designed to support next-generation display and networking technologies.

- The low-profile AGP4X slot supports Digital Video Interface (DVI) to support digital CRT display technology.
- The low-profile Communication and Networking Riser (CNR) slot supports a variety of OEM configuration and expansion options, including support for LAN, modem, audio, and Bluetooth* wireless connectivity.

Thermal and Power

In addition to the system's small footprint, a front-to-back airflow design enables the chassis to be positioned horizontally as a desktop system or to rest on its edge in a "micro-tower" orientation.

The sealed chassis design helps make the system simple, and increases stability and reliability for business and home PC applications.

In place of an internal power supply, the Deep Forest design features a 25V, 120W external Power Brick.

Power management features include support for Suspend-to-RAM and ACPI 1.0.b.

Summary

The Deep Forest Concept PC is a technology demonstration. Although not a product for sale, it proves that Intel's highest level of computing power can be integrated in a small form factor chassis that meets the criteria of the Ease of Use Initiative.

Legacy reduction includes elimination of internal I/O and expansion slots. To save space, the power supply is implemented via an external Power Brick.

Deep Forest integrates Low Profile AGP4X for rich graphics, seven USB slots for easy plug and play expansion, and Instantly Available Power Management for appliance-like operation. It's optimized for networking with integrated 10/100 Ethernet.

When it comes to processing performance, ease of use, and flexible expandability, the Deep Forest Concept PC is the shape of things to come. It points the way toward a new generation of sleek, high-performance PCs with the features and performance to handle the demands of the Visual Internet and the next generation of e-Business applications.

More Info

Visit Intel's Developer Web site for more information on the Ease of Use initiative, legacy reduction in Concept PCs, and Intel's Innovative PC recognition program.

Also see Intel's Developer site for additional information on Digital Video Interface technology and the Communication and Networking Riser specification.

Author Bio

Russ Campbell joined Intel in 1995. He is a member of the Advanced Desktop Platform team responsible for developing Concept PCs in order to demonstrate the latest Intel® technologies and industry initiatives. Prior to joining this team, Russ was the technical marketing engineer for Intel's Desktop Systems Development group working on industry enabling of desktop PC form factors. His engineering background is in storage devices both in desktops and high-performance RAID subsystems.

Intel® ACPI Component Architecture ASL Compiler

Robert Moore
Senior Software Engineer
Technology and Research Labs
Intel Corporation

Overview

The Advanced Configuration and Power Interface (ACPI), an open industry specification originally co-developed and introduced by Intel Corporation, Microsoft Corporation, and Toshiba America Information Systems in December 1996, defines a flexible interface that supports reliable power management through improved hardware and operating system coordination. ACPI allows new power management technology to evolve independently in operating systems and hardware, enabling easier implementation of power management on notebook, desktop and server PCs.

A component of the Intel® Instantly Available PC initiative, ACPI enables the PC to automatically turn on and off peripherals such as CD-ROMs, network cards, hard disk drives, and printers. BIOS developers use an ACPI Source Language (ASL) compiler to implement translation into ACPI Machine Language (AML) to create the ACPI firmware tables for these features.

To support the Instantly Available PC Initiative, Intel developed and released an OS-independent ACPI reference implementation in source code form–the ACPI Component Architecture (ACPI-CA). ACPI-CA provides much of the required infrastructure to locate the ACPI tables, build an ACPI namespace, and manage resource usage information. The ACPI subsystem requires only a small interface layer to retarget it to virtually any operating system.

Although the same group of industry leaders, joined by Compaq Computer Corporation and Phoenix Technologies, released an updated ACPI specification in 2000, a fully functional ASL compiler supporting it hasn't been introduced until now. As part of the Intel ACPI-CA effort, the Intel® ASL compiler provides full support for the new ACPI 2.0 specification and the new ASL grammar elements.

Major Features of the Intel ASL Compiler

In addition to supporting all the ACPI 2.0 specification-defined ASL grammar elements, the Intel ASL compiler provides:

- Extensive syntax and semantic error checking, especially in the area of control methods. This reduces the number of errors that are not discovered until the AML code is interpreted. (Compile-time error checking reduces the number of run-time errors.)
- Multiple types of output files, including formatted listing files with intermixed source, AML, and error messages.
- Portable source code (ANSI C) that allows the compiler to be generated and executed under multiple operating system execution environments.
- The Intel ASL compiler fully supports existing ACPI 1.0 ASL source files. Enhanced compiler error checking will often uncover previously unknown problems in these files.

Compilation

General ASL Syntax Analysis. Multiple errors and warnings are reported in one compile–the compiler recovers to the next ASL statement upon detection of a syntax error. Constants larger than the target data size are flagged as errors. For example, if the target data type is a BYTE, the compiler will reject any constants larger than 0xFF (255). The same error checking is performed for WORD and DWORD constants.

General Semantic Analysis. All named references to objects are checked for validity. All names (both Namepaths and 4-character Namesegs) must refer to valid declared objects.

Control Method Semantic Analysis. Method local variables are checked for initialization before use. All locals (LOCAL0–LOCAL7) must be initialized before use. This prevents fatal run-time errors for uninitialized ASL arguments. In addition, method arguments are checked for validity. For example, a control method defined with one argument can't use ARG4. Again, this prevents fatal run-time errors for uninitialized ASL arguments.

For all ACPI reserved control methods (such as _STA, _TMP, etc.), both the number of arguments and return types (whether the method must return a value or not) are checked. This prevents missing operand run-time errors that may not be detected until after the product is shipped.

Control Method Invocation Analysis. All control method invocations (method calls) are checked for the correct number of arguments in all cases, regardless whether the method is invoked with argument parentheses or not (e.g., both ABC( ) and ABCD). This prevents run-time errors caused by non-existent arguments.

Control methods and invocations are also checked to ensure that if a return value is expected and used by the method caller, the target method actually returns a value. Control method execution paths are analyzed to determine if all return statements are of the same type. This ensures that either all return statements return a value, or all do not. It includes an analysis to determine if execution can possibly fall through to the default implicit return at the end of each method. A warning is issued if some method control paths return a value and others do not.

Output Files

The Intel ASL compiler produces the following files:

- *AML output file*
- *Listing file*–with source file line number, source statements, and intermixed, generated AML code. Include files named in the original source. ASL files are expanded within the listing file.
- *Hex output file*–can be included in a C source file as a table initialization statement.
- *Namespace output file*–shows the ACPI namespace that corresponds to the input ASL file (and all include files).
- *Debug parse trace output file*–gives a trace of the parser and namespace during the compile. Used to debug problems in the compiler, or to help add new compiler features.

Environments Supported

The ASL compiler currently runs on Microsoft Windows* and Linux*. Portable code requires only Bison, Flex, and ANSI C. Source code is distributed with the compiler binaries, and is provided under appropriate license from Intel. Developers can download Intel ASL compiler source code from the Intel Developer Web site.

Summary

Intel's ASL compiler provides BIOS developers with a fully functional development tool that supports all of the new ACPI 2.0 specification-defined ASL grammar elements. It provides greatly enhanced error checking over existing ASL compiler implementations, thus shortening the time required to develop ACPI system firmware.

More Info

For more information on ACPI, check out the ACPI Web site.

Author Bio

Robert Moore joined Intel in 1982. He worked as a systems software engineer and senior software engineer in the iRMX/Multibus group on the iRMX 86, iRMX II, iRMX III, and iRMX for Windows operating system products before joining the Mobile Architecture Lab. He earned a B.S. degree in Computer Engineering from the University of Michigan, and an M.S. degree in Computer Science from the University of Illinois.

Networking and Communications

Embedded Intel® Architecture Caching Solutions

Dave Hillyard
Communications Platform Manager
Embedded Intel Architecture Division
Intel Corporation

Overview

The increasing popularity of the Internet continues to drive rapid growth in data networking. While the Web supports a vast amount of data traffic, a large percentage of this traffic is redundant. This is due to the fact that multiple users at any given location typically request much of the same information at any given time. As a result, a significant portion of Wide Area Network (WAN) traffic consists of identical request and data content.

Web caching is a technique designed to optimize available bandwidth and improve overall network performance by storing frequently accessed information closer to the user. In addition to its network performance benefits, Web caching provides an opportunity to reduce network communication costs. Several vendors are developing caching appliances based on Intel® Architecture hardware, readily available software, and standardized interface components to achieve substantial cost savings and shorter time-to-market.

Caching

Caching is not a new concept. For years computers have used caching to improve system performance by storing frequently used information closer to the processor. At the network level, caching can be defined as a technique for keeping frequently accessed information in a location closer to the requester. Because of its physical or logical proximity to the user, the cache storage location can provide faster access compared with retrieving data from across the network.

All cache servers work basically the same way. They intercept object requests traveling from a browser to a Web server, and then store the requests on a hard drive as the objects travel back from the Web server to the browser. In this way, subsequent requests for the same object by other browsers can be intercepted by the cache, which then returns the object from its own memory rather than forwarding the request to a remote server. Ideally, object requests fulfilled by the cache should save both time and communications bandwidth.

Client Caching

Browser applications allow an individual to cache Web pages (images and HTML text) on the client's local hard drive. The ability to cache Web objects at the browser has been available for some time. This method serves single users well, but it provides no benefit for other users on the same network who may be accessing the same Web sites.

Network Caching

In order to optimize the use of available network bandwidth and benefit multiple users, caching can be implemented at the network level. For example, network caching is currently widely used by Internet Service Providers (ISPs) who wish to increase the performance level of their service. On the consumer side, faster connectivity solutions such as Digital Subscriber Lines (DSL) and cable modems are enhancing Web browsing performance, which is in turn increasing the traffic load on the network backbone.

At the same time, content providers are diversifying and enriching data types with more complex formats such as streaming audio/video and Java* applets. These richer data types tend to increase the bandwidth requirement of each individual Internet session. These trends are forcing ISPs to look for cost-effective ways to enhance the existing communications and networking infrastructure, while keeping pace with customer demand. Network caching will continue to be a big part of their solution.

According to "The 1999 Internet Caching Report" by the Internet Research Group, the total size of the caching market is projected at nearly $2.2 billion in 2003 with an increasing number of vendors and users.

Network Caching Methods

Network caching can be implemented in two ways: as a proxy or in transparent mode. Early caches were typically implemented on general-purpose servers running proxy caching as an application along with many other applications.

Proxy caches are intrusive in accepting requests and passing them on to their destinations. Every Web client must be configured to direct all outgoing calls to the proxy's IP address. The proxy checks to see whether it has cached the valid reply for a requested object. If it has not cached the valid reply, the proxy makes a decision to forward the request to the destination server. In addition, a proxy may also implement security firewall, user authentication, traffic administration, content filtering, and a host of other functions.

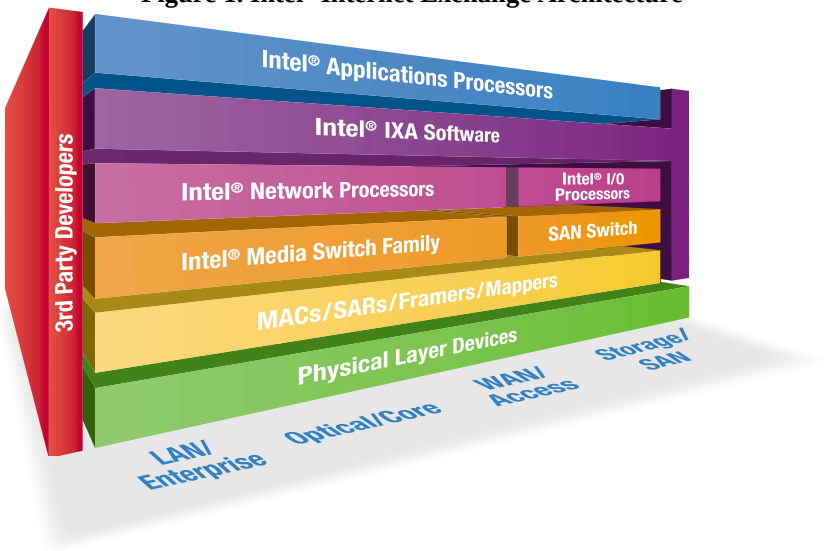
Although proxy caches help in value-added services, they slow down Web access, and any failure of the proxy server causes all users to lose network access. The cache server can also be overloaded and add an incremental performance limitation.

A transparent cache hangs off the network near the Internet gateway and invisibly intercepts the traffic as it passes from the browser to the Web server. The transparent cache uses a policy-based router or a Layer-4 switch that diverts the HTTP traffic to the cache server or a group of servers. A truly transparent caching solution should be flexible, support scalability, and incorporate adequate load balancing among multiple cache servers. It should also be working in "fail-safe" mode in the event that one or all the cache servers become unavailable. Transparent caching eliminates the need to configure individual browsers to point to the cache, while reducing administrative overhead.

Intel® Internet Exchange Architecture

Intel supports caching at all levels of network infrastructure with Intel® Internet Exchange Architecture (IXA). Intel® IXA, shown in Figure 1, provides a consistent framework for OEMs and independent software vendors to quickly deploy new networking and communications services and develop differentiated networking products that deliver scalable performance with reduced total cost of ownership. Intel IXA includes end-to-end development solutions and building blocks that enable developers to create solutions for the entire Open Systems Interconnectivity (OSI) stack.

Figure 1. Intel® Internet Exchange Architecture

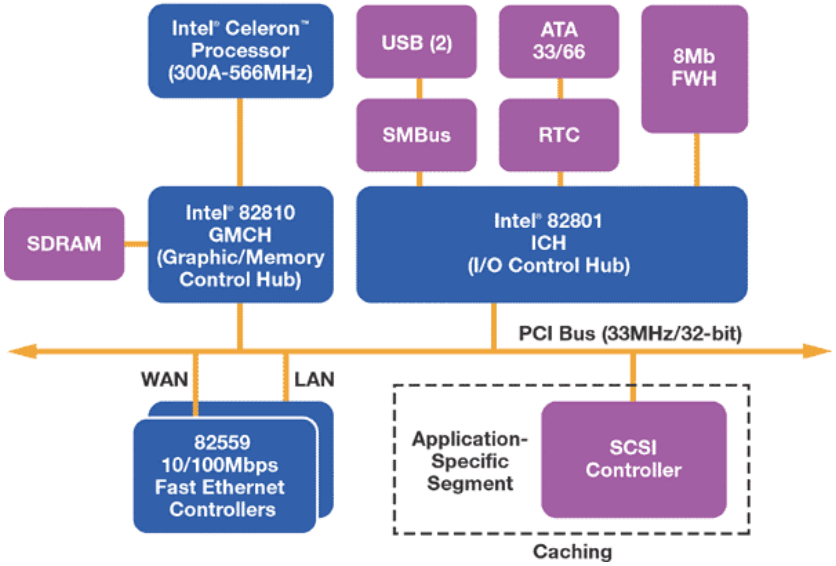


Embedded Intel Architecture delivers solutions that meet the performance requirements of the Application Services Layer of the OSI Model. By incorporating scalable embedded Intel Architecture components and software within Intel IXA, Intel is delivering a flexible top-to-bottom architecture. This architecture provides high performance, scalability, and code compatibility, along with programmability that enables more cost-effective software-based product differentiation.

Intel® Appliance Reference Design

Figure 2 is a block diagram of a typical appliance board based on embedded Intel Architecture components including an Intel® Celeron™ processor, Intel® 810 chipset, the Intel® 82559 Ethernet controller with Intel® flash memory, and SDRAM components. This free reference design provides a fast time-to-market development platform that can be used as a stand-alone integrated Internet appliance for caching applications.

Figure 2. Embedded Intel® Architecture Value Communications Appliance Block Diagram



Summary

With the convergence of voice and data and the increased use of the Internet in enterprise network environments, work groups, and small office/home office settings, network caching is becoming a powerful and cost-effective tool to improve network performance. Adding cache appliances based on embedded Intel Architecture to the existing network at critical points can reduce network latency and improve access efficiency while keeping WAN implementation costs under control. Easy installation, maintenance, and configuration are features that make network caching appliances attractive in enhancing the value of the Internet.

More Information

Visit Intel's Developer Web site for more information on:

- Intel Internet Exchange Architecture
- Value Communications Appliance Reference Design
- Intel Celeron Processor

Author Bio

Dave Hillyard directs communications platform strategy activities for Intel's Embedded Intel Architecture Division. He joined Intel in 1989, and for the last several years he has been involved with emerging communications services and products, including hardware, software, and system-level programs and initiatives. Before joining Intel, Dave served in various system-level engineering capacities at Motorola, Inc. and Digital Equipment Corp.

—End of Intel Developer Update Magazine Issue 16—